

一、计算机基础知识

1、JS [JavaScript 参考 - JavaScript | MDN \(mozilla.org\)](#)

JS在整个H5课程体系里面非常重要，举足轻重；

- HTML 用于控制网页的结构
- CSS 用于控制网页的外观
- JavaScript 控制着网页的行为

2、计算机的组成：

- CPU：大脑，中央处理器，处理指令；
- 内存：程序在运行时是需要一些数据的，这些数据是放在内存里面的；
- 硬盘：数据存储硬盘里面；
- 输入设备：鼠标、键盘、触摸屏、麦克风、摄像头等
- 输出设备：显示器、屏幕、音响等；
- 人和人交互：自然语言
汉语：
英语：
- 这些自然语言都有一个环境；

3、人和机交互：编程语言

- JAVA、Go、Python、C++、.Net、C#、JavaScript (JS解释器) 等；
- 机和机交互：协议
- HTTP (Browser/Server) 、TCP、UDP、FTP
- 我们写程序就是写指令：交给CPU执行的指令（语句）；
- JS：全称是JavaScript
- 和Java的关系：
JAVA：雷锋；
JavaScript：雷峰塔；

4、一个页面分成三个层：*****

- 结构层：HTML，负责提供内容的；骨
- 表现层：CSS，负责修饰页面的，也就是页面效果；肉
- 行为层：JS，负责页面交互，赋予页面灵魂；魂
提升用户体验；
减轻服务器压力；

JS的更多使用场景：

- JS可以进行服务器开发：Nodejs、Egg；
- 搭建客户端开发环境：Proxy（代理）、webpack；

JS学习方法：形成一个完整的知识架构

127学习方法：

- 10%自学：百度、CSDN；
- 20%跟别人学：课程、B站；
- 70%靠自己实践：总结、项目、给别人讲、分享；

二、JS: JavaScript

1、运行环境：JS解释器

浏览器：

- 浏览器里面集成的有JS解释器；
- 如果你想实现一些效果，需要用到浏览器提供的API
- 比如获取一个标签并修改他的样式；

Nodejs：

- 也集成了V8（chrome的js解释器）这个JS解释器；
- 如果你想进行服务器开发，就会用到nodejs提供的相关的API；
- 比如http；
- 学习环境：在浏览器里面执行我们的JS；

2、如何引入JS代码到HTML页面

1. 行内：类似于行内样式，保密（怕你用行内的方式引入JS代码）；

- 可以将单行或少量JS代码写在HTML标签的事件属性中(以on开头的属性)，如： onclick
- 注意单双引号的使用：在HTML中我们推荐使用双引号，JS中我们推荐使用单引号
- 可读性差，在 HTML 中编入 JS 大量代码时，不方便阅读

2. 特殊情况下使用

内嵌：使用script标签引入代码片段；

- 可以将多行JS代码写到<script>标签中
- 内嵌 JS 是学习时常用的方式

3. 外联：使用script标签的src属性来引入指定的js文件；

- 利于HTML页面代码结构化，把单独JS代码独立到HTML页面之外，既美观，又方便
- 引用外部JS文件的script标签中间不可以写代码
- 适合于JS代码量比较大的情况

3、输出数据到控制台

- alert() 主要用来显示消息给用户
- document.write() 浏览器窗口输出
- console.log() 用来给程序员看自己运行时的消息
- console.log()：打印常规信息到控制台；

- console.dir(): 打印更完整的信息到控制台;
- console.info(): 打印普通信息到控制台;
- console.warn(): 打印警告信息到控制台;
- console.error(): 打印错误信息到控制台;

4、JS分成【三部分】：

1. ES: 语法部分; ECMAScript: ES

- ECMA:欧洲计算机制造商联盟; JS遵守ECMA执行的规范: ES;
- JS是运行在客户端的(目前);
- ES就是语法部分; JS核心技术这个阶段主要就是学习ES;

2. DOM: 结合上浏览器提供的接口来完成页面效果实现

文档对象模型(Document Object Model, 简称DOM), 是W3C组织推荐的处理可扩展标记语言的标准编程接口。通过 DOM 提供的接口可以对页面上的各种元素进行操作(大小、位置、颜色等)。

- DocumentObjectModel 文档对象模型
- 用来操作页面的, 比如修改标签的样式、内容等;

3. BOM: 必须通过浏览器提供的相关接口才能完成浏览器操作

BOM (Browser Object Model, 简称BOM) 是指浏览器对象模型, 它提供了独立于内容的、可以与浏览器窗口进行互动的对象结构。通过BOM可以操作浏览器窗口, 比如弹出框、控制浏览器跳转、获取分辨率等。

- BrowserObjectModel 浏览器对象模型;
- 用来操作浏览器的; 刷新、跳转等;

三、页面的效果实现:

1.事件:

概念: 鼠标点击事件、输入事件、拖动事件等, JS可以捕捉到页面上发生的所有的东西;

2.事件三要素:

- 事件源(谁)
- 事件类型(什么事)
- 事件处理程序(做啥)

(1)事件源 事件被触发的对象 谁 按钮

(2) 事件类型 如何触发 什么事 比如鼠标点击(onclick) 还是鼠标经过 还是键盘按下

(3) 事件处理程序 通过一个函数赋值的方式 完成

3.执行事件的步骤:

- 获取事件源

- 注册事件(绑定事件)
- 添加事件处理程序(采取函数赋值形式)

4.语法:

```
事件源.事件 = function () {  
    事件处理函数  
}
```